

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF MASSACHUSETTS

CONNECTU LLC,

Plaintiffs,

v.

MARK ZUCKERBERG, EDUARDO SAVERIN,
DUSTIN MOSKOVITZ, ANDREW MCCOLLUM,
CHRISTOPHER HUGHES, and FACEBOOK,
INC.,

Defendants.

CIVIL ACTION NO. 1:04-CV-11923
(DPW)

MARK ZUCKERBERG, and FACEBOOK, INC.,

Counterclaimants,

v.

CONNECTU LLC,

Counterdefendant,

and

CAMERON WINKLEVOSS, TYLER
WINKLEVOSS, and DIVYA NARENDRA,

Additional Counterdefendants.

**DECLARATION OF JAMES BUTTERWORTH IN SUPPORT OF THE
FACEBOOK, INC.'S RESPONSE TO CONNECTU'S RENEWED FORENSIC
RECOVERY ARGUMENTS**

I, James Butterworth, declare as follows:

1) I am currently employed by Guidance Software, Inc. as the Manager of Professional Services and a practicing Computer Forensics Consultant. Attached hereto as Exhibit A is my *curriculum vitae*. Guidance is the maker of EnCase Forensic software. EnCase Forensic is the industry standard in computer forensic investigation technology. I make this declaration in support of The Facebook, Inc.'s Response to ConnectU's Renewed Forensic

Recovery Arguments. I make this declaration of my own personal knowledge, and if called as a witness, I could and would testify competently to the truth of the matters set forth herein.

2) I have 20 years of service to the United States Navy with over 13 years of hands-on experience in Computer Network Security, covering all aspects of vulnerability testing, perimeter defense installations, and Intrusion Detection techniques. I have provided professional computer forensic consulting services for private investigators and local attorneys. I also have performed permissive and court ordered non-permissive seizure, data extraction and replication, emergency recovery, and forensic analysis for criminal cases, domestic abuse investigations, child exploitation investigations, preserving evidence and providing definitive results by using professional licensed forensic software and employing industry standard MD5 algorithm verification methods.

3) On March 15, 2006, a paralegal from the Orrick law firm hand delivered to me 9 original electronic devices. These devices are identified as:

- a) 371-02 Maverick Server
- b) 371-03 Maverick Server
- c) 371-04 Andrew McCollum's hard drive
- d) 371-05 Andrew McCollum's Dell Inspiron 8200
- e) 371-06 Dustin Moskovitz's ThinkPad
- f) 371-07 Andrew McCollum's Sony Vaio
- g) 371-08 Andrew McCollum's Kingston Thumb Drive
- h) 371-09 Dustin Moskovitz's Maxtor hard drive
- i) 371-10 Andrew McCollum's WD Caviar hard drive

On March 18, 2006, the same paralegal hand delivered to me Device Nos. 371-01 (Mark Zuckerberg's Sony Vaio) and 371-11 (Dustin Moskovitz's Sony Vaio).

4) Guidance was retained by The Facebook to perform a broad collection of data from the 11 devices described above. We reviewed the results of Mr. Berryhill's work and determined that he produced approximately 1GB of data. In addition to the volume of production, I have reviewed the type of files that Mr. Berryhill produced using EnCase version

3.22g, which has survived Daubert/Frye, and it is my opinion that Mr. Berryhill's production was reasonable, given that he was only tasked to look for and retrieve files with extensions ".pl, .php, .phpt, .htm, and .html". During our review, we were tasked with identifying and retrieving those same file extensions, in addition to ".sql and .mdb" file extensions. Our production volume is approximately 26.4 Gigabytes, a very large percentage of which is attributable the ".sql" file extension, which Mr. Berryhill did not produce. To date, our fees for this project total \$22,870.50.

5) Device No. 371-01 was excluded from our analysis because we were unable to image the hard drive; we tried three different methods to acquire a duplicate digital image. We employed EnCase 5.05a natively using Fastbloc write blocking device. We attempted to acquire it using EnCase's Linux product, known as LinEn, with Fastbloc write blocking device. And, finally, we used Helix Linux distribution with "DD" imaging software and fastbloc write blocking device. In all three instances, the drive would indicate it was "spinning up," and would allow at least a rudimentary search, but would not allow for an entire bit stream copy.

6) We analyzed a total of 10 device images (EnCase .E01-.En files) using EnCase 5.05a. The following steps were taken in this regard:

a) Each image was scanned for deleted files and folders using the "Recover folders" option in EnCase. This process scans the unallocated space looking for both folders and files, and upon locating them, places them inside a "Lost Folders" folder within EnCase. This process ensures that deleted files are then searched.

b) Files with extensions .tar, .gz, .zip, .tgz were mounted recursively to be able to access the files within these compound files. These files were searched. We found

they contained source code, which is in the set of files we provided to Facebook.

c) We conducted a link file analysis to search for all .lnk files to determine if files were copied to removable media. Our search revealed zero instances where a link file contained a reference to removable media. There were a few links that referenced a digital camera, but the files referenced were reviewed and did not contain any source code.

Linux machines do not have .lnk files.

d) A condition (filter) was created to identify all code files with the following criteria:

- Files with created date of December 31st 2004 or earlier;
- File with extension being either .pl, .php, .phpt, .htm, .html, .sql, or .mdb;
- Files that have .sql, .php as part of their name (like abc.php.bak);
- Files that are either not deleted or deleted but not overwritten on disk.

The condition was run across the images and identified files were exported out keeping the exported folder structure intact, wherever possible. The metadata including created date, last modified date and last written date for identified code files was exported to a report per device image. These results are contained in comma separated value spreadsheets.

e) A condition (filter) was created to identify all files having extensions of either .txt or .doc and created on or before December 31st 2004. Metadata for these files was exported to a report per device image.

f) Our process saves the full path of files, including metadata.

7) I read Plaintiff's Summary of March 3, 2006 Status Report (Docket No. 148). Plaintiff contends that Facebook did not do all that ConnectU would have done to locate

and recover code. *Summary*, pg. 8. Plaintiff also writes that Mr. Berryhill's work was sloppy and that he is not EnCase certified. *Id.* Even if true, it does not follow that the EnCase software used by Mr. Berryhill did not work as designed. The use of EnCase software, even by someone who is not EnCase certified, does not change data. In fact, in my experience, courts routinely find that the use of EnCase software by someone who has used it successfully before is reliable. We reviewed the files produced to Plaintiff as a result of Mr. Berryhill's forensic analysis. Although Plaintiff complains that Mr. Berryhill did not search for .tar, .tgz and .bak files, files contained within these compressed archives can be found in the files produced to Plaintiff as a result of Mr. Berryhill's work.

8) Plaintiff writes that Mr. Berryhill's "search strategy would also miss signature mismatches (i.e., files in which the file extension has been changed from the extension automatically provided by the program, such as changing ".PHP" to ".TXT")." *Summary*, pg. 9. File signature becomes important because the headers of binary files are fixed, *i.e.*, the same data exists at every file offset in the same file header. For example, a .jpg file will have the same field in File Offset 7 as any other .jpg file. The files that Plaintiff asks to have searched are not, however, binary files; they are text format files. Text formatted files do not have defined, uniformly accepted file headers. Therefore, conducting a file signature analysis/comparison against such files is not reliable and would not likely lead to the recovery of additional code.

9) Plaintiff writes that Facebook did not provide the database definitions file. *Summary*, pg. 11. A database definition file acts or, more appropriately, *can act*, as a reference file for how a database should create the tables. A database definition file is not a binary file format. In fact, there is no standard to follow. During the programming process, the programmer may elect to have the program execute the generation of a database based upon the

table design contained with the database definition file. This is a user created document, not something created as a result of having a database. In my opinion, to adequately search for a database definition file on the submitted computer media, Plaintiff should first define the database definitions that it is asserting are at issue in this matter. The process is far better served by searching for a unique string in their definition tables and then using that string as a keyword to search the media.

10) Plaintiff writes that Facebook did not search .doc or .txt files for database definitions. *Summary*, pg. 9. As described above, database definitions do not necessarily exist. In my opinion, therefore, if Plaintiff believes it will find its database definitions on Facebook computers, Plaintiff should provide a list of those definitions to be searched. Otherwise, looking for generic “database definitions” is likely a fruitless effort. Moreover, to the extent Plaintiff suggests that Facebook should search all .txt and .doc files for code, that too should be rejected. Random searches of these files would be incredibly time consuming and expensive and, as with the database definitions, is not likely to produce the results Plaintiff expects. Instead, Plaintiff should provide a list of keywords or terms for Facebook to search across the .txt or .doc files. We can perform such a search relatively quickly and efficiently.

11) Plaintiff writes that it would search for “files and fragments that were not recovered or restored.” *Summary*, pg. 11. File Slack analysis is normally used in criminal cases where the existence of a digital artifact then becomes the proof that a certain activity has taken place, *i.e.*, a fragment of a victim’s bank account number is carved out of file slack, when the suspect claims no knowledge of the victim. The existence of a bank account number of the victim on the suspect’s computer may be able to link them, as this information would not end up in file slack as a normal case of operation. In other words, that binary bit pattern did not just

suddenly “form up” on the suspect’s computer. In my experience, information found in slack or unallocated space is not part of eDiscovery, as file slack is exclusively fragmented and unallocated space is highly circumstantial. In other words, a file may exist, but there may be no other information about it that will shed light on file activity. Analysis of file slack is burdensome, expensive and, virtually always, fruitless in civil matters. Having completed forensic examinations on literally thousands of computers myself, I offer the following opinion. The assertion that a fragment of a file located in the unallocated (*i.e.*, unused area) space of a drive, or potentially be located in file slack of an unrelated file, is flawed for the following reason. A file fragment is simply that, a fragment. An examiner will be unable to piece the fragments together. The tool that our company, Guidance Software, makes, sells, and is currently in use by a high percentage of all computer forensic examiners in the world, attempts to do this automatically by parsing through various table records and analyzing the file extent. Without knowing the file extent, there is simply no way an examiner would be able to link these fragments together.

12) Plaintiff writes that it is necessary for it to have images of the hard drives (rather than copies) in order to verify the respective “file creation, last-modified, and last-accessed dates” of certain folders or files and to identify “directory creation dates.” *Summary*, pg. 11. Plaintiff’s argument for requiring this information is not based in fact. A folder is its own entity, having its own attributes — a user can copy files and folders within it, from it and to it. Depending on the method used to do such copying, the attributes of the file folder may or may not change. Plaintiff’s statement that because the “coursename” and “courseparse” files have December 2003 dates, then the folder in which they reside also must have been created by at least that date, is not accurate. The user could have created the folder yesterday and placed

files created in 2003 inside the folder without changing the file attributes (*e.g.*, the creation date) of the folder. If Plaintiff identifies the files for which it wants metadata, we can easily provide that information. The file attributes for the coursename and courseparse files are as follows:

Name:	coursename.pl
File Ext:	pl
Description:	File, Archive
Is Deleted:	No
Last Accessed:	02/02/06 01:04:41 AM
File Created:	02/02/06 01:04:41AM
Last Written:	12/23/03 04:25:11 PM
Entry Modified:	11/03/05 01:06:17 AM
File Acquired:	03/16/06 04:35:58 PM

Name:	courseparse.pl
File Ext:	pl
Description:	File, Archive
Is Deleted:	No
Last Accessed:	02/02/06 01:04:41 AM
File Created:	02/02/06 01:04:41AM
Last Written:	06/26/04 02:17:41 AM
Entry Modified:	11/03/05 01:06:17 AM
File Acquired:	03/16/06 04:35:58 PM

13) Plaintiff writes that it would have identified “computer and program activity dates ... including dates and times of usage and uploading of Harvard Connection and facebook.com code to the server. *Summary*, pg. 13. The server, “Maverick,” is a Linux-based server and does not journal activity like a Windows-based event log or have file attributes that operate in the same way as the Windows Operating System. Even in Windows (which the Maverick server is not), the likelihood that these file attributes would still be on the server is unreasonable, unless there had been no file access or modifications made to the file(s), since the time a file was uploaded to the server. This includes the requirement that there were no file backups, no virus scans, or any other applications that have the capability to open a file as a matter of normal operation.

14) Plaintiff writes that it would have confirmed “authorized user identifications,” and identified the “original complete directory structure.” *Summary*, pg. 13. In

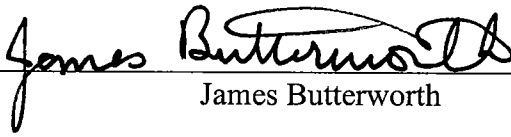
my opinion and experience, this information is entirely irrelevant to a search for software code and generally is not part of an ediscovery order.

15) Plaintiff notes that “ConnectU also found a file that deletes old versions of code.” *Summary*, pg. 4. On Device No. 371-02, which is a server used by Facebook, a program called “Subversion” exists. Subversion is version control software used by software developers to maintain versions of software. The existence of the program on a software server is expected and, in my opinion, is not evidence of an improper intent to delete files.

16) I reviewed the contents of four CDs labeled C011338, C005288, C011339, and C011143. These CDs have a label on them, representing that they are the “Sourcecode produced by CONNECTU on (date).” I reviewed these CDs using the same criteria that we are using to cull through the devices provided by Facebook. At best, there is only 171.69 MB of information, spanning the four CDs. They appear to be nothing more than a back up of the Harvard Connection website (HCSite.zip), along with some other SQL table dumps and some folders containing entire copies of websites. There is no reference to a) recovery of deleted items, b) file fragments, or c) searching in file slack. In fact, all of these files are archived in their corresponding folders. One CD, C011338, has folders that were created on August 15, 2005, two weeks prior to the CDs being produced. In my opinion, these four CDs do not represent a forensically sound production. On another CD, C005288, there is a single file on the root of the CD, entitled “Connectu_sourcecode.txt,” which has a file created date of June 2, 2005 at 5:44 P.M. Incidentally, this is the same time the CD was prepared, indicating that this process did not preserve the original files or metadata — the very thing that Plaintiff complains Facebook’s production lacks.

I declare under penalty of perjury that the foregoing is true and correct to the best

of my knowledge. Executed this 29th day of March, 2006.


James Butterworth